



ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Caio Emídio Daniel
Geovana Silva Villafranca
Jhonata Ferreira de Oliveira
Rafael Tomazelli Lopes
Vinicius Ruffo Viviani
Vitor Alves Martins

RA 821136207
RA 820143966
RA 821124383
RA 821120761
RA 821156267
RA 817122770

PinnochData – Sistema para controle de pedidos

SÃO PAULO

Formatado: À esquerda

Sumário

1. Introdução.....	3
1.1 Tema.....	3
1.2 Objetivos a serem alcançados.....	3
1.3 Escopo principal.....	3
2. Definição do modelo de processo.....	3
3. Requisitos do Sistema de software.....	3
3.1 Requisitos funcionais.....	3
3.2 Requisitos não funcionais.....	4
4. Casos de uso.....	5
5. Projeto.....	6
5.1 Arquitetura lógica.....	6
5.2 Arquitetura física.....	7
6. Protótipo de interface.....	8
7. Critérios de qualidade de software.....	9
8. Testes.....	9
8.1 Plano de testes.....	9
8.2 Roteiro de testes.....	12
9. Anexos.....	12
9.1 Roteiro de entrevistas.....	12

1. Introdução

1.1 Tema

Software de controle de pedidos.

1.2 Objetivos a serem alcançados

Este software tem como objetivo alcançar um público voltado para empresas do ramo de varejo, que será desenvolvido na plataforma NetBeans IDE 12.0 e relacionamento com o banco de dados MySQL Server. Sendo assim, vamos realizar em foco automatização de processos na importação de pedidos de venda.

1.3 Escopo principal

Como principal objetivo será destacado o gerenciamento para se obter um melhor desempenho para conseguir almejar uma melhor performance no gerenciamento de pedidos.

2. Definição do Modelo de Processo

Neste projeto decidimos utilizar o método SCRUM como a melhor opção para gerenciar o nosso desenvolvimento do software.

3. Requisitos do Sistema de Software

3.1 Requisitos Funcionais

RF01 - O sistema deve possibilitar ao usuário listar todos os pedidos que estão disponíveis no momento.

RF02 - O sistema deve possibilitar o usuário a filtrar os produtos por categorias, cor, marca e tamanho.

RF03 - O sistema deve possibilitar ao usuário classificar os produtos por preço, relevância e data de lançamento.

RF04 - O sistema deve possuir a funcionalidade do usuário alterar o status dos pedidos

RF05 - O sistema deve consultar o status atual de um pedido quando o usuário solicitar.

RF06 - O sistema deve elaborar relatórios entre datas inseridas pelo usuário.

RF07 - O sistema deve ter a opção do usuário se cadastrar no sistema utilizando um email e preenchendo suas informações.

RF08 - O sistema deve permitir o usuário a se logar no sistema.

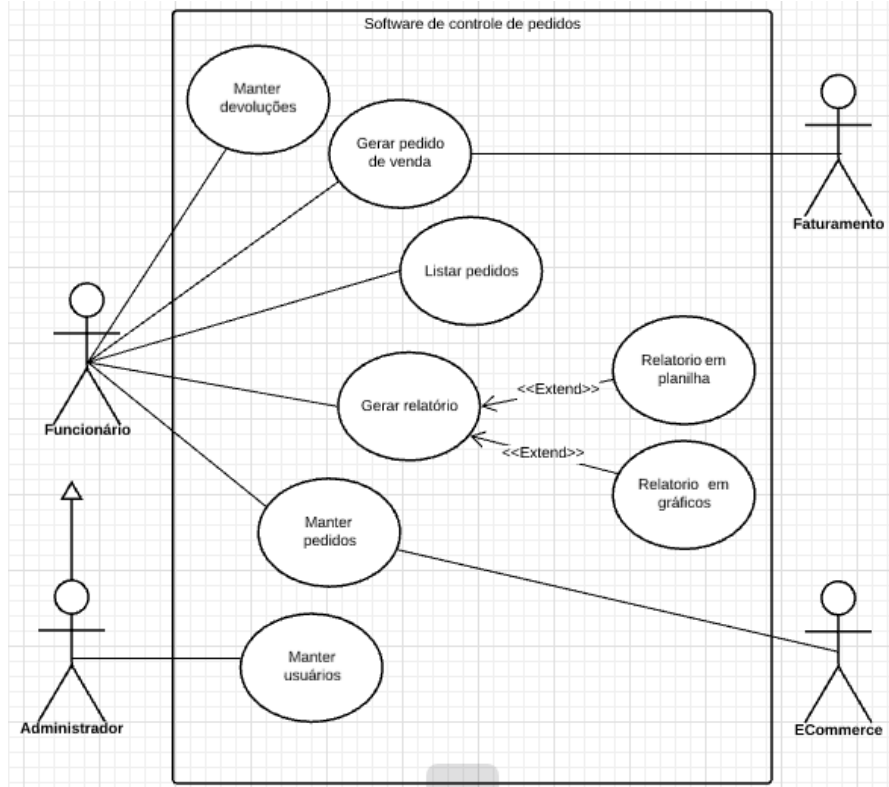
RF09 - O sistema deve possibilitar o usuário a realizar reembolsos de pedidos.

3.2 Requisitos Não-Funcionais

- RNF 01** - O sistema deve ser compatível com os sistemas operacionais Linux, Windows e Mac.
- RNF 02** - O sistema deve ter conexão com um banco de dados MySQL.
- RNF 03** - O sistema deve ocupar menos que 2GB.
- RNF 04** - O sistema necessita de uma conexão com internet de no mínimo 2mb.
- RNF 05** - O sistema deve ser desenvolvido na linguagem de programação JAVA.
- RNF 06** - O sistema requer no mínimo o processador i3 3220/ AMD FX 6300.
- RNF 07** - O sistema requer no mínimo 2GB de memória RAM.
- RNF 08** - O sistema requer um processador e sistema operacional de 64 bits.
- RNF 09** - O sistema deve conter autenticação com usuário e senha válidos (cadastrados no Banco de dados) para permitir que o usuário comece a utilizar o sistema.
- RNF 10** - A senha do usuário deverá ser gravada utilizando o algoritmo SHA-3 para criptografia.
- RNF 11** - O sistema deve ter alta disponibilidade como 85% do tempo.
- RNF 12** - A interface do sistema deverá se comportar adequadamente independente do sistema operacional que o usuário estiver.
- RNF 13** - Deverá notificar o usuário caso o mesmo se mantenha inativo por mais de 30 minutos.
- RNF 14** - Deverá emitir uma mensagem de erro caso o usuário não esteja com uma conexão com internet adequada.

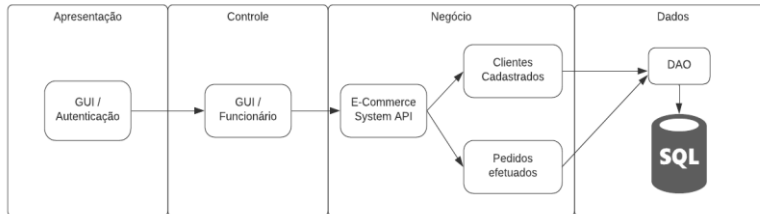
4. Casos de Uso

Os requisitos funcionais devem ser representados por meio de modelo de caso de uso.



5. Projeto

5.1 Arquitetura Lógica



Escolhemos a arquitetura em camadas devido ao fácil entendimento, facilitação de sua manutenção e o rápido desenvolvimento do software. Além do já citado uma das vantagens que a arquitetura em camadas possui é criar uma hierarquia de níveis de modos de acesso, protegendo as camadas mais internas e possibilita que uma camada trabalhe com diferentes versões de outra camada.

A linguagem escolhida foi Java, pois contém uma versatilidade entre as plataformas. O Java é orientado a objetos, seguro e robusto, tendo uma variedade de conteúdo para aplicações. Dentro da equipe todos tem conhecimento sobre a linguagem.

O banco de dados escolhido foi o MySQL, pois se trata de um banco de dados relacional bem confiável e com alta disponibilidade, que possui uma grande comunidade de usuários e é muito disseminado. Sendo um banco de dados confiável e um dos mais conhecidos, toda nossa equipe possui conhecimento com esse banco de dados.

Incluiremos componentes externos como APIS para o sistema de faturamento dos pedidos e sistema de e-commerce.

Um dos padrões de projetos que podem ser aplicados, é o padrão comportamental Command (Também conhecido como "Comando"), consiste no objeto que é utilizado para encapsular toda informação que é necessitada para realizar uma ação ou um evento em um momento posterior, permitindo introduzir novos comandos na aplicação sem quebrar o código cliente existente, este padrão dará ganhos em usabilidade e manutenibilidade.

Outro padrão que pode ser utilizado é o Adapter, é um padrão de projeto estrutural que permite objetos com interfaces incompatíveis colaborarem entre si, pois o nosso sistema trabalhará com muitos dados e objetos de sistemas diferentes, podendo haver a necessidade de adaptar os mesmo a um certo padrão que utilizaremos nos processos dos sistemas, isso nos permite uma maior

portabilidade e eficiência para o sistema, o que contribui para a comunicação entre eles.

E não podemos esquecer de utilizar o padrão Construtor (Também conhecido como Builder), onde é um padrão muito importante, pois ele permite que um código produza diferentes tipos e representações de um objeto, tendo uma ótima funcionalidade.

5.2 Arquitetura Física

Qual a configuração de hardware será adequada para a utilização do software? Justifique.

R: Para que o sistema consiga oferecer toda sua funcionalidade sem ocorrer algum travamento e tendo em vista que terá outros aplicativos em aberto, será necessário para no mínimo uma configuração:

Processador i3 3220/ AMD FX 6300;

2GB de memória RAM;

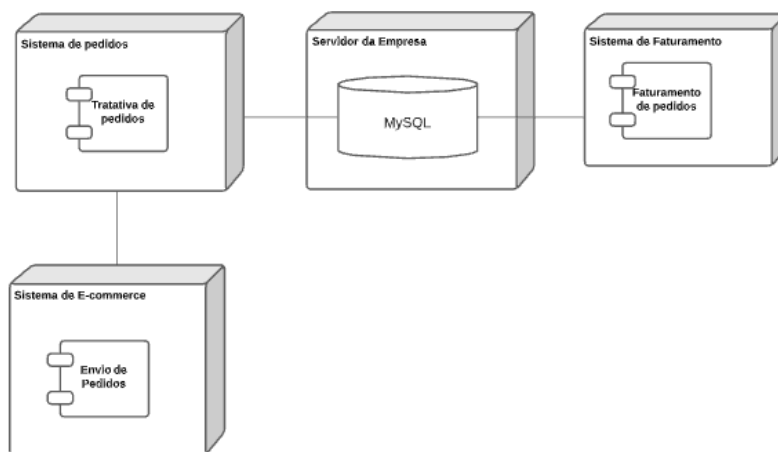
Capacidade mínima de memória 5 GB

Qual a configuração de rede será adequada para a utilização do software?

Justifique.

R: O sistema necessita de uma conexão com internet de no mínimo 2mb para que o usuário consiga navegar de uma maneira estável.

Inclua um diagrama de implantação (opcional).



6. Protótipo de Interface

PinnochData Opções Ajuda

Emissão de relatórios

Selecione o conteúdo do relatório:

- ☐ Pedidos cancelados
- ☐ Pedidos concluídos
- ☐ Pedidos devolvidos

Selecione as respectivas datas para início e fim do relatório

Data de início: 00 / 00 / 0000

Data final: 00 / 00 / 0000

Gerar relatório

Protótipo baseado no caso de uso “Gerar relatório”, as opções de relatório em planilha ou em gráficos aparecem após clicar no botão “Gerar relatório”.

PinnochData Opções Ajuda

Pesquise pelo código do pedido, data de emissão ou nome do cliente

Cód	Emissão	Cliente	Pagamento	Entrega	Total
#35	10/11/2021	José Carlos	Aprovado	Enviado	R\$195,00
#36	10/11/2021	Maria Cláudia	Aprovado	Enviado	R\$1.000,00
#37	10/11/2021	Antônio Luiz	Aprovado	Enviado	R\$1.250,00
#38	10/11/2021	Mário Borges	Aprovado	Enviado	R\$456,60
#39	09/11/2021	Claúdio Gilberto	Aprovado	Enviado	R\$199,99
#40	09/11/2021	Marcos Paulo	Aprovado	Enviado	R\$506,85
#41	09/11/2021	Antônio Nunes	Negado	Enviado	R\$1.950,90
#42	09/11/2021	Gilberto Gil	Negado	Enviado	R\$4.520,45
#43	09/11/2021	Antônio Fagundes	Cancelado	Enviado	R\$690,99

Protótipo baseado no caso de uso “Listar pedidos”.

7. Critérios de Qualidade de Software

Usabilidade:

O software deve ser de fácil operação e de fácil entendimento aos conceitos principais do software, para que todos os funcionários possam facilmente operá-lo de imediato sem necessidade de treinamento prévio.

Eficiência:

O Software deve se mostrar eficiente, realizando suas funcionalidades de forma rápida e correta, sem comprometer o desempenho. É de extrema importância o software ser mais eficiente que o software anterior, trazendo assim maiores benefícios para a empresa.

8. Testes

8.1 Plano de Testes

Neste item deve ser criado o plano de testes do sistema, permitindo a validação do sistema por parte do desenvolvedor, através da verificação dos requisitos do sistema desenvolvido.

Referência: UC Gestão e Qualidade de software

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se todos os pedidos pendentes são listados e se eles estão corretos.

Requisitos que motivaram este teste: RF01 - O sistema deve possibilitar ao usuário listar todos os pedidos que estão disponíveis no momento.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Pré-condições: estar logado, haver pedidos novos entrando na base de dados.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o sistema consegue filtrar os produtos por categoria, cor, marca e tamanho.

Requisitos que motivaram este teste: RF02 - O sistema deve possibilitar o usuário a filtrar os produtos por categorias, cor, marca e tamanho.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o usuário consegue classificar os produtos por preço, relevância e data de lançamento.

Requisitos que motivaram este teste: RF03 - O sistema deve possibilitar ao usuário classificar os produtos por preço, relevância e data de lançamento.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o usuário consegue alterar o status dos pedidos.

Requisitos que motivaram este teste: RF04 - O sistema deve possuir a funcionalidade do usuário alterar o status dos pedidos.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o sistema consulta o status atual de um pedido quando o usuário solicitar.

Requisitos que motivaram este teste: RF05 - O sistema deve consultar o status atual de um pedido quando o usuário solicitar.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o sistema elabora relatórios entre datas inseridas pelo usuário.

Requisitos que motivaram este teste: RF06 - O sistema deve elaborar relatórios entre datas inseridas pelo usuário.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o usuário consegue se cadastrar no sistema utilizando um e-mail e preenchendo suas informações.

Requisitos que motivaram este teste: RF07 - O sistema deve ter a opção do usuário se cadastrar no sistema utilizando um e-mail e preenchendo suas informações.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Pré-condição: O usuário precisa ter um e-mail funcional.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o usuário consegue se logar no sistema.

Requisitos que motivaram este teste: RF08 - O sistema deve permitir o usuário a se logar no sistema.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Pré-condição: O usuário precisa ter uma conta registrada no sistema.

Tipo de teste: Funcional

Subtipo de teste: requisitos

Objetivo do teste: Testar se o usuário consegue realizar reembolso de pedidos.

Requisitos que motivaram este teste: RF09 O sistema deve possibilitar o usuário a realizar reembolsos de pedidos.

Técnica para criação dos casos de teste: Baseado em casos de uso.

Pré-condição: O usuário precisa estar logado e um pedido deve estar feito.

Tipo de teste: Não Funcional

Subtipo de teste: ambiente de execução

Objetivo do teste: Testar se o sistema está rodando corretamente nas plataformas especificadas

Requisitos que motivaram este teste: RNF 01 - O sistema deve ser compatível com os sistemas operacionais Linux, Windows e Mac.

Técnica para criação dos casos de teste: técnicas de caixa preta

Critério de sucesso: Todos os testes funcionais devem ser executados em todas as plataformas apresentando sucesso.

Recursos necessários: Máquinas virtuais e físicas com as plataformas especificadas.

Tipo de teste: Não Funcional

Subtipo de teste: dependência

Objetivo do teste: testar a conexão do sistema com o banco de dados

Requisitos que motivaram este teste: RNF 02 - O sistema deve ter conexão com um banco de dados MySQL.

Técnica para criação dos casos de teste: técnicas de caixa preta

Pré-condição: Banco de dados MySQL com os dados do sistema.

Tipo de teste: Não Funcional

Subtipo de teste: Disponibilidade

Objetivo do teste: testar a disponibilidade do sistema

Requisitos que motivaram este teste: RNF 11 - O sistema deve ter alta disponibilidade como 85% do tempo.

Técnica para criação dos casos de teste: testes de sobrecarga.

Pré-condição: Software específico de teste de sobrecarga.

Tipo de teste: Não Funcional

Subtipo de teste: confiabilidade

Objetivo do teste: Requisitos que motivaram este teste: RNF 09 - O sistema deve conter autenticação com usuário e senha válidos (cadastrados no Banco de dados) para permitir que o usuário comece a utilizar o sistema.

Técnica para criação dos casos de teste: testes de Segurança.

Pré-condição: usuário cadastrado, solução específica para teste de segurança.

8.2 Roteiro de Testes

Clique no link > [Roteiro de Testes](#)

9. Anexos

9.1 Roteiro entrevista

Fale sobre quais são as suas principais atividades na organização (o entrevistado falará sobre suas tarefas e responsabilidades) e/ou explique como é atualmente o processo do escopo principal.

R: O Gerente comercial informa que ele e sua empresa, necessita de um software para gestão do gerenciamento de pedidos no ramo de varejo, onde o mesmo encontra dificuldades no desempenho do processo atual.

Quais são as vantagens e desvantagens do processo atual na organização?

R: O processo atual tem vantagens por conta de sua interface de fácil entendimento e manuseio, como desvantagens o entrevistado informa que é necessário importar o pedido do sistema para realizar alterações e importá-lo novamente.

Apresentar ao entrevistado o ESCOPO PRINCIPAL (item 1.3)

Quem poderia usar este produto na organização? Justifique.

R: Apenas a equipe de gerenciamento de pedidos irá utilizar o software.

Quais necessidades este produto atenderá? Justifique.

R: Atenderá a necessidade de ter um processo mais ágil e eficaz. Devido ao alto custo dos funcionários a empresa necessita de um sistema mais ágil para redução de funcionários, diminuindo os gastos da empresa.

Quais os objetivos deste produto? Justifique.

R: O principal objetivo é aumentar a automatização do processo atual para obter mais performance na área.

Como este produto adicionará valor para a empresa? Justifique.

R: Devido ao processo mais ágil e eficaz que o software pretende propor, a empresa pode gerar mais lucros e menos gastos.

Quais atributos deste produto são críticos para atender às necessidades da empresa e portanto, para o sucesso do projeto? Justifique.

R: O software deve alterar pedidos, cancelar pedidos e excluir pedidos, além de consultar o status de pedidos ele também necessita de uma interface de rápido entendimento e fácil manuseio.

Como o produto proposto se compara aos produtos existentes dos concorrentes ou da mesma empresa? O que você sabe sobre produtos similares? Justifique.

R: O produto é similar aos já existentes no mercado, porém necessita ser melhor do que o software atual utilizado pela empresa. O entrevistado informa que os concorrentes utilizam um software bem automatizado, onde não necessita realizar importação e exportação dos pedidos.

Quais os critérios de qualidade para o produto (software) proposto? Justifique.

R: Velocidade, confiabilidade, usabilidade e funcionalidade.

Quais sugestões você teria para melhorar o produto proposto? Justifique.

R: A sugestão proposta seria que todos os processos de alteração de pedido sejam feitos diretamente no sistema, sem necessitar da utilização de softwares externos.

Qual a sua expectativa de tempo de entrega para o produto, levando em consideração todos os pontos e necessidades a serem atendidas?

R: O entrevistado tem a necessidade do software ser entregue em 6 meses.

Você pretende utilizar em apenas uma plataforma (mobile, desktop) ou gostaria de algo híbrido?

R: Apenas desktop e deve poder ser utilizado em qualquer sistema operacional.

Qual função adicional você acha que poderia haver no software?

R: O entrevistado gostaria que o sistema elaborasse relatório de pedidos concluídos, cancelados e de devoluções entre datas que o usuário especificaria.

Você teria algo a acrescentar sobre o que conversamos?

Sim, o entrevistado mencionou que tem uma grande dificuldade com o estoque disponível onde há falta de mercadorias então o mesmo mencionou que gostaria de uma inteligência vinda diretamente dos pedidos, onde proporciona uma projeção de vendas, sendo assim, podendo contribuir com a equipe comercial de vendas e a equipe de estoque.

Clique no link da gravação da entrevista: <https://youtu.be/z9Dltbn8l0>